

CLAIMS

1. (previously amended) A method of scheduling the handling of data from a plurality of channels, comprising:
- accumulating data from a plurality of channels by a remote access server, at respective predetermined input rates;
 - providing data of each of the plurality of channels, at respective predetermined output rates;
 - scheduling a processor of the server to handle the accumulated data from at least one first one of the channels, without interruption, once during a first cycle time, defined by the respective input and output rates of the first channels; and
 - scheduling the processor to handle the accumulated data from at least one second one of the channels, without interruption, once during a second cycle time different from the first cycle time, the second cycle time being defined by the respective input and output rates of the second channels.
2. (Original) A method according to claim 1, wherein the first cycle begins concurrently with a second cycle.
3. (Original) A method according to claim 2, wherein the first cycle time is an integer multiple of the second cycle time.
4. (Original) A method according to claim 1, wherein scheduling the processor to handle the accumulated data comprises scheduling the processor, during the second cycle, to handle the accumulated data from substantially all the at least one second channels, before scheduling the processor to handle data from any other of the plurality of channels.
5. (Original) A method according to claim 4, wherein scheduling the processor to handle the accumulated data from the at least one first one of the channels comprises checking whether the second cycle has elapsed and scheduling the processor to handle the accumulated data from one of the at least one first channels only if the second cycle has not elapsed.

6. (Canceled)

7. (previously amended) A method according to claim 1, wherein the scheduling comprises scheduling the processor to handle the accumulated data from at least one of the second channels at least twice before scheduling the processor to handle data from at least one of the first channels.

8. (Original) A method according to claim 1, wherein scheduling the processor to handle the accumulated data comprises allowing the processor to utilize up to a predetermined amount of processing time for each channel.

9. (Original) A method according to claim 1, wherein the processor does not run an operating system which performs preemption.

10. (Original) A method according to claim 1, wherein scheduling the processor comprises having the processor wait without handling data from any of the channels if all the channels were scheduled for handling during their respective current cycles.

11. (Original) A method according to claim 10, comprising measuring the waiting time of the processor in the first cycle and using the measured time in determining whether to accept handling data from an additional channel.

12. (Canceled)

13. (Original) A method according to claim 1, comprising processing an entire block of accumulated data of the scheduled channel responsive to the scheduling.

14.-16. (Canceled)

17. (Previously amended) A remote access server, comprising:

a plurality of channel drivers which accumulate data from respective channels, at respective predetermined input rates and provide data of each of the plurality of channels, at respective predetermined output rates;

a processor which handles the accumulated data; and

a scheduler which schedules the processor to handle accumulated data from a first channel once during a first cycle time, defined by the timing of the driver of the first channel, and data from a second channel once during a second cycle time different from the first cycle time, without interrupting the processor while it is processing data from a channel.

18. (Original) A server according to claim 17, wherein the scheduler schedules the processor to handle the data from the first channel at least twice before scheduling the processor to handle data from the second channel.

19.-25. (Canceled)

26. (currently amended) A method of scheduling the handling of data, by a remote access server keeping track of a short cycle and a long cycle, from a plurality of channels including at least one short cycle channel and at least one long cycle channel, comprising:

accumulating data from the plurality of channels by the server;

in each short cycle, scheduling a processor of the server to handle the accumulated data from all the short cycle channels, before scheduling the handling of data of any long cycle channel;

determining whether a current short cycle has elapsed after scheduling the processor to handle the data from all the short cycle channels; and

scheduling the processor to handle the accumulated data from one of the at least one long cycle channel if the current short cycle did not elapse according to the determining, if there is a long cycle channel which was not scheduled yet during the current long cycle.

27. (original) A method according to claim 26, comprising determining whether the current short cycle has elapsed after scheduling the processor to handle the data from the long cycle channel, and scheduling the processor to handle the accumulated data from an additional long cycle channel, if the current short cycle did not elapse.

28. (original) A method according to claim 26, comprising waiting, after scheduling the processor to handle the data from all the short cycle channels, until the beginning of the next short

cycle without processing data from any channel, if all the long cycle channels were already scheduled during the current long cycle.

29. (original) A method according to claim 26, wherein the long cycle begins concurrently with a short cycle.

30. (original) A method according to claim 26, wherein the long cycle time is an integer multiple of the short cycle time.

31. (currently amended) A method of scheduling the handling of a plurality of connections, comprising:

accumulating data from a plurality of ~~channels~~ connections requiring handling at respective predetermined rates, by a remote access server;

determining for at least one of the connections a quality of service level; and

scheduling the processor to process data from the plurality of connections in an order determined responsive to the determined quality of service level.

32. (original) A method according to claim 31, wherein the scheduling comprises scheduling the processor to handle data from at least one first connection before handling data from at least one second connection having a lower quality of service level than the at least one first connection.

33. (original) A method according to claim 31, comprising changing the quality of service level of at least one of the connections while accumulating the data and changing the order of scheduling responsive to the change in the quality of service level.

34. (Previously added) A method according to claim 1, wherein if a channel is not processed in its respective cycle the channel suffers from starvation.